Psychtoolbox 3.0

Routines for Psychophysics

Outline

- Hardware and Software
- PTB installation
- Testing and Timing
- Basic functions
- Examples and Exercises
- Miscellaneous

Hardware and Software

- Requirements
- Displays

    LCD

    CRT

    Projectors

- Matlab, GNU/Octave

Hardware and Software

Basic functionality available on any graphics hardware:

Basic OpenGL 1.0 support required.

16 MB VRAM required.

High speed and advanced features:

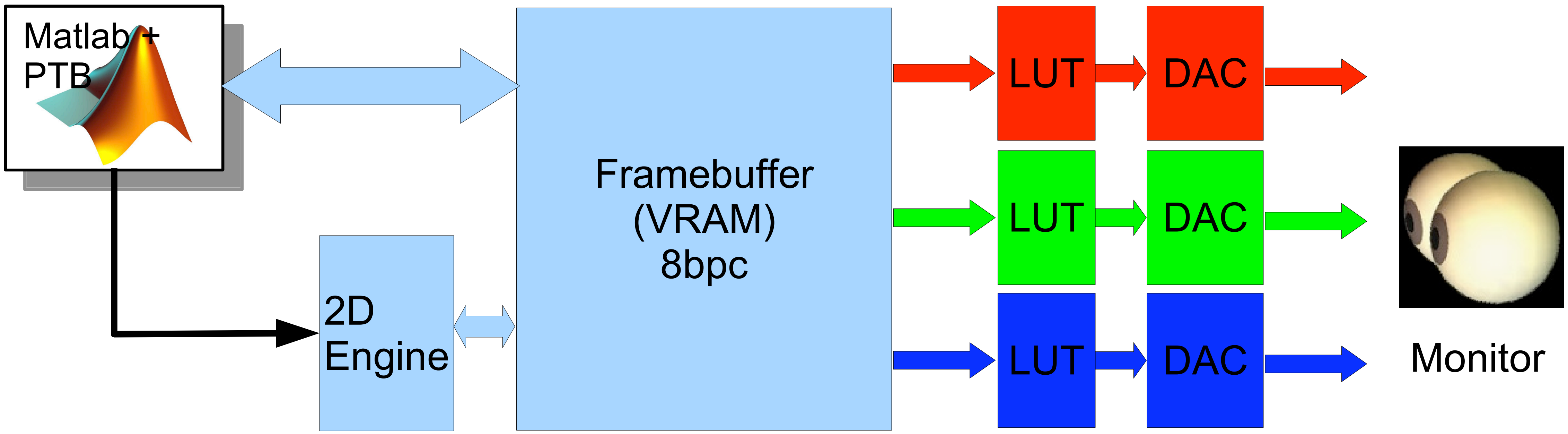
OpenGL 1.5, better 2.0/2.1 capable.

> 128 MB VRAM, more is better.

Simplifies a lot of complex stimulus programming tasks. Allows for high precision, e.g., 32 bpc floating point.

PTB-3 regularly tested on: NVidia GeForce, ATI Radeon. Onboard chips are very limited in functionality and speed! (e.g. Intel MacBooks, iMacs)

Old Graphics Hardware

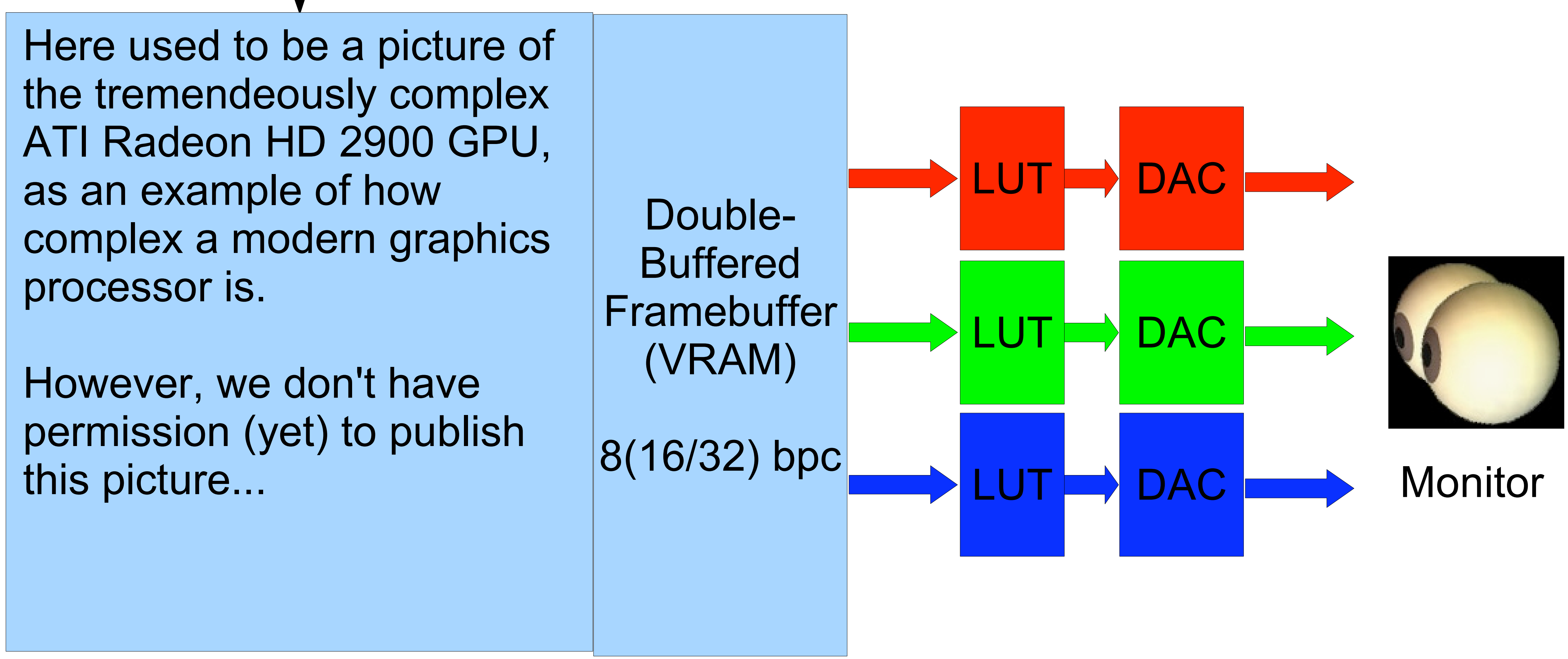


- Graphics card == Mostly passive image store.
- Psychtoolbox draws directly into Framebuffer while buffer is scanned out for display.
- Most drawing operations implemented/executed in software on slow CPU.
- Only very few operations are hardware-accelerated (Image copy, filled rectangles).

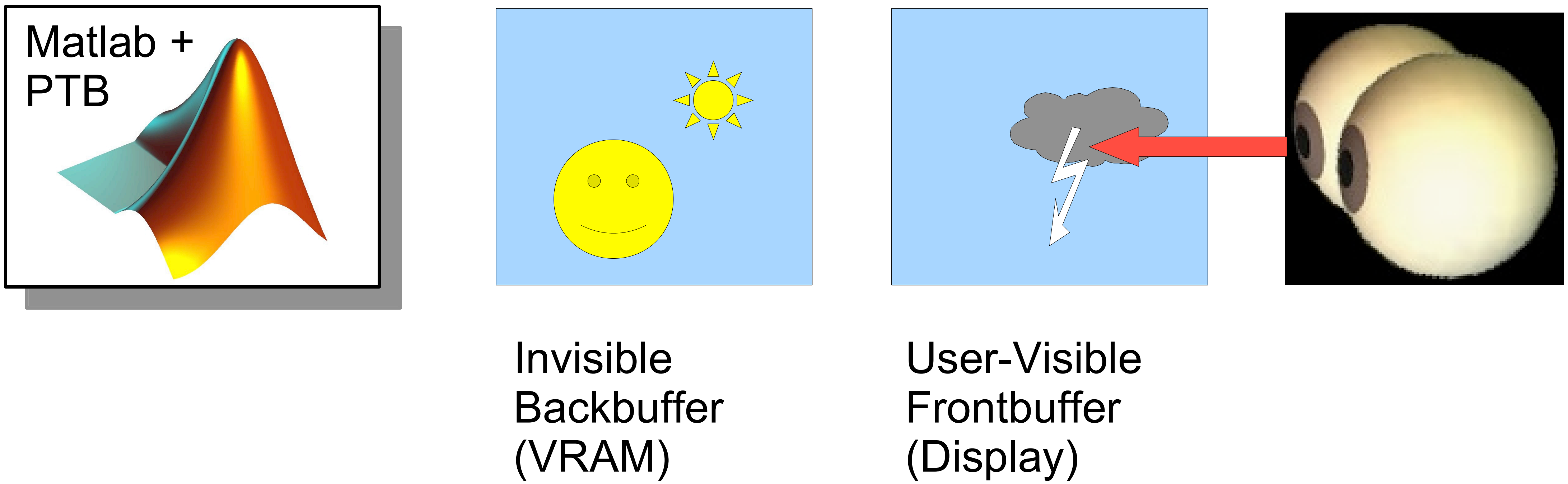
Modern Graphics Hardware

Matlab + PTB

- GPU's are massive parallel and highly programmable.
- High memory bandwidth: > 100 GB/s vs. 8 GB/s on CPU.
- Computational speed: 475 GFlops vs. 12-24 GFlops.
- PTB interfaces with the GPU mostly via OpenGL.

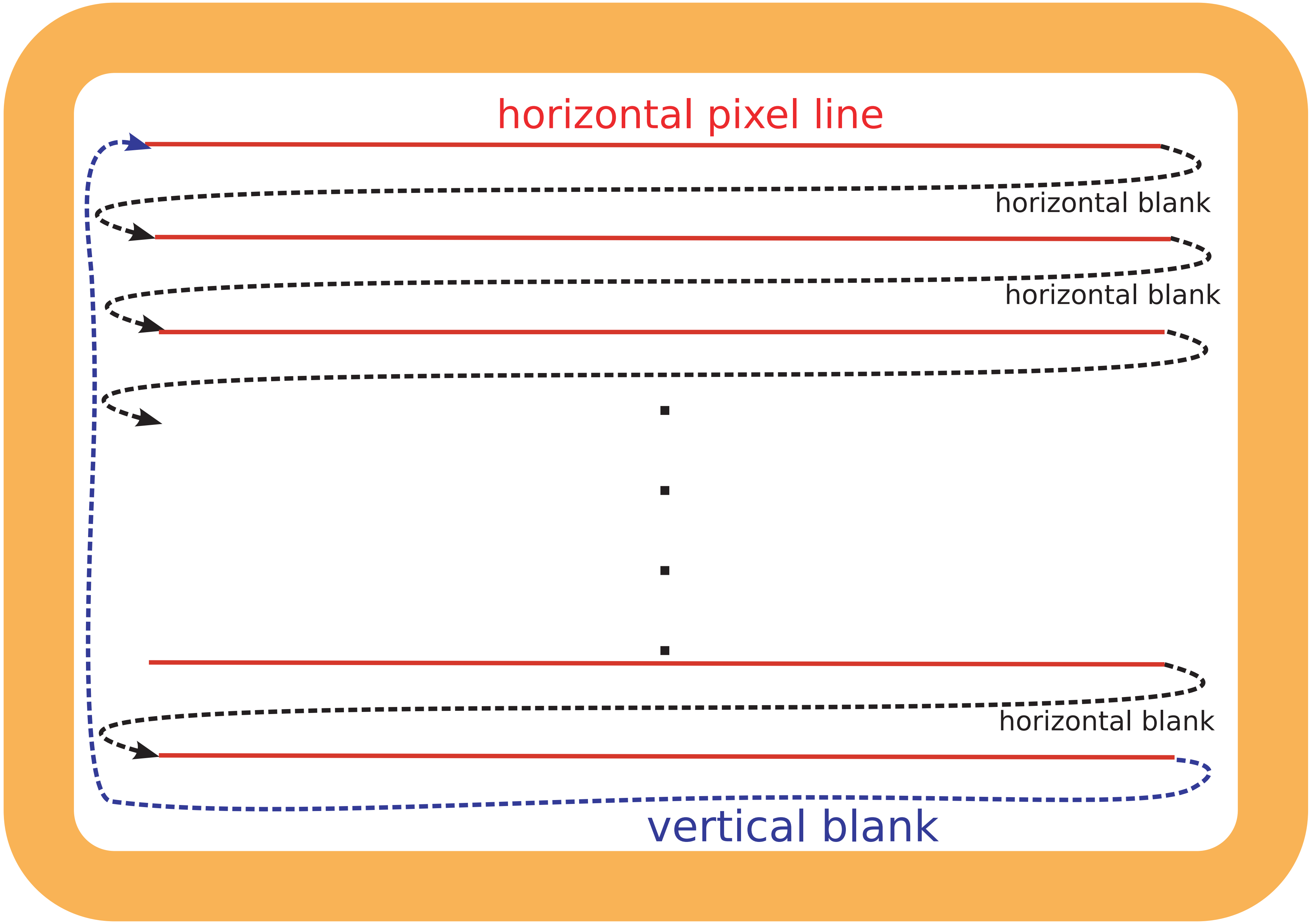


PTB-3: Double Buffered Drawing



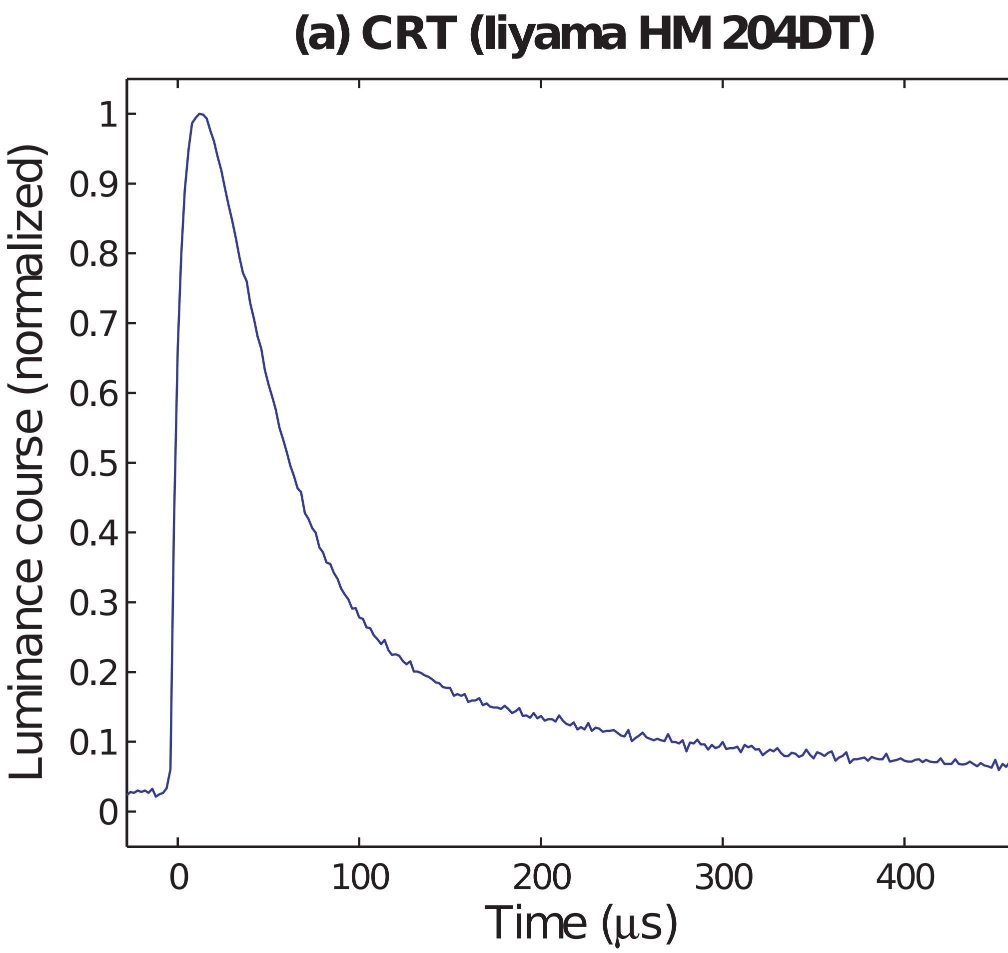
1) Subject sees cloudy picture on the display screen (==Frontbuffer).
2) Matlab issues Screen() drawing commands.
3) Graphics hardware draws into backbuffer, processing the OpenGL commands in the background while Matlab and Psychtoolbox are able to do other stuff in parallel, e.g., keyboard queries, sound output...

CRT Display



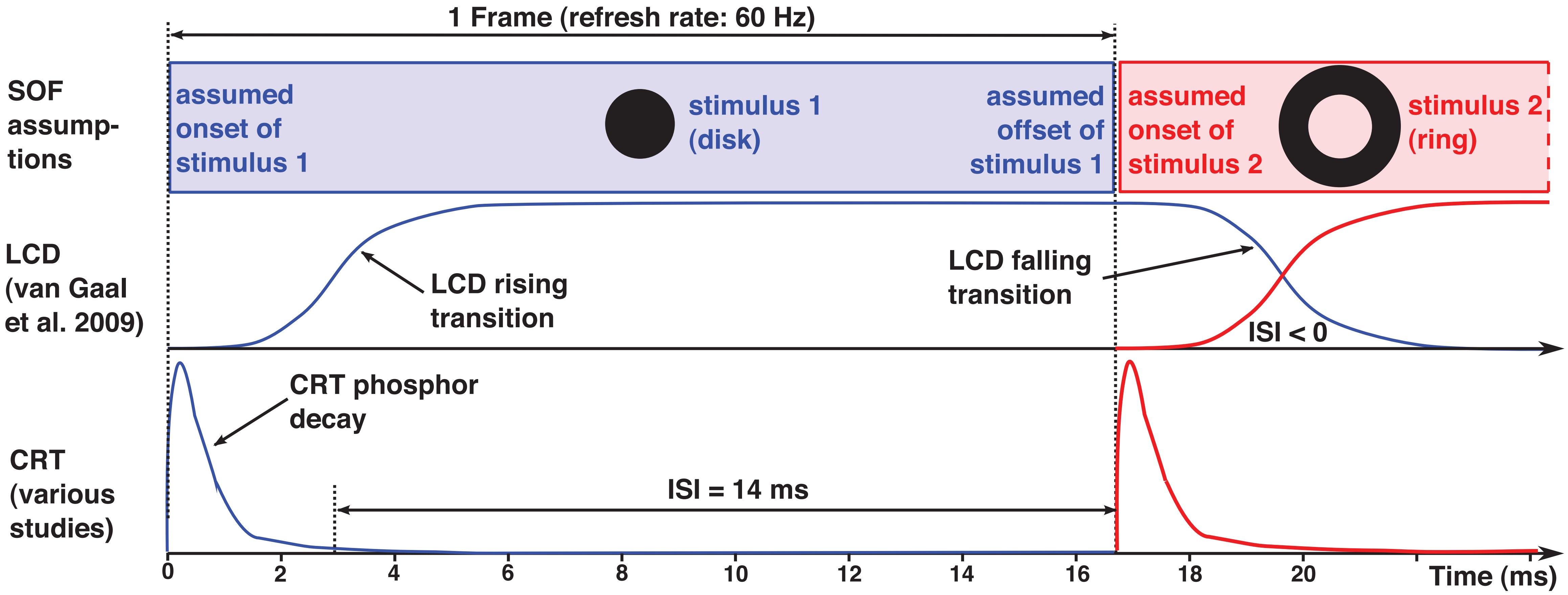
Elze (2010) *PLoS ONE*

CRT vs LCD



Elze (2010) *PLoS ONE*

CRT vs LCD



Elze (2010) *PLoS ONE*

PTB Installation

- for Windows
- for Mac OS/X
- for Linux

Basic Functions

- `OpenWindow()`
- `Screen()`
- `try - catch`
- `Priority()`
- `AssertOpenGL`
- `VBLSyncTest`

Screen()

Controls all aspects of the graphics- and display-hardware.

- Performs all 2D drawing operations.
- Controls stimulus onset timing and provides timestamps.
- Allows for some high performance image processing.
- Performs on-demand stimulus post-processing.

try-catch-end

```
   try
   % main prg

catch
   % catch error: This is executed in case something goes wrong
   % in the'try' part due to programming error:

   % Do same cleanup as at the end of a regular session...
   Screen('CloseAll');
   ShowCursor;
   fclose('all');
   Priority(0);

   % Output the error message that describes the error:
   psychrethrow(psychlasterror);

end
```

Screen()

```
screens=Screen('Screens');
screenNumber=max(screens);

gray=GrayIndex(screenNumber);

[w, wRect]=Screen('OpenWindow',screenNumber, gray);
Screen(w,'BlendFunction',GL_SRC_ALPHA, GL_ONE_MINUS_SRC_ALPHA);

Screen('Flip', w);

tex=Screen('MakeTexture', w, imdata);

Screen('DrawTexture', w, tex);

[VBLTimestamp startrt]=Screen('Flip', w);

Screen('CloseAll');
```

Priority()

```
% Returns as default the mean gray value of screen:
gray=GrayIndex(screenNumber);

% Open a double buffered fullscreen window on the screen
% 'screenNumber' and choose a gray background. 'w' is the handle
% used to direct all drawing commands to that window - "Name" of
% the window. 'wRect' is a rectangle defining the size.
% See "help PsychRects"
[w, wRect]=Screen('OpenWindow',screenNumber, gray);

% Set priority for script execution to realtime priority:
priorityLevel=MaxPriority(w);
Priority(priorityLevel);
```

KbCheck

```
% Make sure keyboard mapping is the same on all operating systems
% Apple MacOS/X, MS-Windows and GNU/Linux:
KbName('UnifyKeyNames');

% assign response key
resp=KbName('b'); % respond with key 'b'

% initialize KbCheck and variables to make sure they're
% properly initialized/allocted by Matlab - this to avoid time
% delays in the critical reaction time measurement part of the
% script:
[KeyIsDown, endrt, KeyCode]=KbCheck;


while ( KeyCode(resp)==0)
      [KeyIsDown, endrt, KeyCode]=KbCheck;
      WaitSecs(0.001);
end
```

Example and Exercises

Example and Exercises

- Display image HolidaySnap.jpg
  `HolidaySnap1.m`

Example and Exercises

- Display image HolidaySnap.jpg
  `HolidaySnap1.m`
- Draw an anti-aliased fixation dot
  `HolidaySnap2.m`

Example and Exercises

- Display image HolidaySnap.jpg
  `HolidaySnap1.m`
- Draw an anti-aliased fixation dot
  `HolidaySnap2.m`
- Record response time and response
  `HolidaySnap3.m`

Micellaneous

- Textures and Alpha Blending
  `AlphaImageDemo.m`
- QuickTime movie playback
  `SimpleMovieDemo.m`
- Image processing pipeline
  `ProceduralGaboriumDemo.m`
- 3D Graphics OpenGL
  `UtahTeapotDemo.m`
- PsychPortAudio

Summary

Three Commandments of Experimentation:

- Know thy stimuli
- Know thy methods
- Know thy observers